

GÖKHAN ARAGÜLÜ

LEAD GAME DEVELOPER | UNREAL ENGINE C++

Gameplay Architecture • Multiplayer • Production Systems • Tools

DREAMER SHARK

PORTFOLIO / GAME TECHNOLOGY

Istanbul, Türkiye • gokhanaragulu@gmail.com • [LinkedIn](#) • [Fab / AFROX](#)

PROFILE

Lead Game Developer specializing in Unreal Engine C++ with nearly a decade of engine experience. Focused on scalable gameplay architecture, multiplayer/replication, performance-conscious systems, editor tooling, and production-ready C++/Blueprint APIs. Combines technical leadership with hands-on implementation across active game development.

EXPERIENCE

Misa Games | Lead Game Developer

May 2025 - Present

- Lead Unreal Engine C++ gameplay architecture and reusable production systems for active game development.
- Define maintainable C++/Blueprint boundaries for multiplayer-aware gameplay, designer iteration, and performance.
- Drive technical integration across gameplay, replication, tools, debugging, and cross-disciplinary production needs.

Sarte Industries | Lead Game Developer

Apr 2023 - May 2024

- Led Unreal implementation across game, VR, and metaverse initiatives, emphasizing reusable interactive systems and production workflows.
- Supported cross-team technical integration while maintaining scalable, maintainable Unreal architecture.

Zonex Entertainment Studios | Lead Game Developer

Nov 2022 - May 2023

- Led gameplay and virtual-world system development in Unreal Engine, delivering reusable real-time systems for production.

YD Yazılım ve Teknoloji | Lead Game Developer

Sep 2022 - Dec 2022

- Led implementation of gameplay logic and interactive systems, strengthening production-oriented Unreal development practices.

Sheriff Games | Lead Game Developer

Jan 2021 - Sep 2022

- Led Unreal Engine gameplay development, technical direction, and reusable project systems across production needs.

SELECTED WORK SNAPSHOT

Baklava Simulator 1 & 2

PUBLISHED PC / STEAM

Gameplay, UI, interaction, save/settings and reusable simulation systems.

AF Inventory Blueprint

PUBLISHED ON FAB

DataTable-driven inventory with drag/drop, pickup/drop and no continuous Tick.

Afrox Dialogue System

PUBLISHED ON FAB

C++ runtime/editor dialogue framework with data-driven authoring and Blueprint APIs.

Project Verdict

IN DEVELOPMENT

Multiplayer investigation game with structured match flow, evidence, voice and automated tests.

CORE STACK

Unreal Engine 5.x

C++ / Blueprints

Gameplay Architecture

Multiplayer / Replication

AI / Gameplay Systems

Editor Tools / Automation

UMG / Slate Integration

Save / Load & Data-Driven Systems

TECHNICAL LEADERSHIP

Architecture & direction

Scalable systems, clear ownership boundaries, maintainable APIs.

Cross-disciplinary delivery

Gameplay, networking, tools, debugging and integration.

Production quality

Profiling, testing, automation and maintainability.

PUBLIC WORK

Fab Marketplace - AFROX

2 published Unreal Engine products

PC / Steam

2 published game titles

LINKS

[LinkedIn](#)

[Fab / AFROX](#)

PORTFOLIO

Dreamer Shark

Professional portfolio and public technical work.

TECHNICAL DEPTH

ARCHITECTURE

Modular gameplay systems, reusable subsystems/components, clear ownership boundaries, data-driven design, save/load orchestration, and testable APIs.

GAMEPLAY & AI

Combat, inventory, dialogue, interaction, targeting, AI perception, player-state flows, HUD/UI integration, and system orchestration.

ANIMATION & CUSTOMIZATION

Motion Warping, Contextual Animation, Motion Matching research, animation-facing gameplay integration, Mutable character-customization workflows.

NETWORKING

Replication-aware architecture, server/client responsibilities, multiplayer flow design, synchronization concerns, and interest management research.

TOOLS & AUTOMATION

Editor tooling, build/test automation, designer-facing Blueprint exposure, debugging workflows, profiling, and maintainability-focused implementation.

PRODUCTION SYSTEMS

Reusable save/settings, audio/presentation services, orchestration, validation, and production-first system ownership across project lifecycles.

SELECTED ENGINEERING WORK

Afrox Dialogue System

FAB - PUBLISHED

C++ dialogue framework with separate runtime/editor modules, branching choices, timed decisions, Rich Text, typewriter behavior, Data Asset authoring, and Blueprint-friendly extension points.

C++ • Data Assets • Editor Module • UMG • Blueprint API

AF Inventory Blueprint

FAB - PUBLISHED

DataTable-driven inventory architecture focused on practical customization, drag/drop workflows, pickup/drop, item actions, and efficient runtime behavior without continuous Tick.

Blueprint • DataTable • Inventory • No Tick

Baklava Simulator 1 & 2

PC / STEAM - PUBLISHED

Gameplay/simulation systems spanning production loops, interaction, UI, presentation, save/settings architecture, audio/presentation services, and reusable production systems.

Simulation • Gameplay • UI • Save/Settings • Architecture

Project Verdict

UNREAL ENGINE - IN DEVELOPMENT

Multiplayer investigation / social-deduction project with deterministic match flow, evidence and contradiction systems, detention/verdict flow, voice architecture, and production-focused automated testing.

Multiplayer • Game Flow • Evidence • Voice • Automation

ENGINEERING LEADERSHIP & DELIVERY

ARCHITECTURE OWNERSHIP

Translate gameplay requirements into maintainable C++/Blueprint architecture with explicit system boundaries and reusable APIs.

PRODUCTION DELIVERY

Optimize for iteration speed without sacrificing profiling, debugging, testing, maintainability, or long-term ownership.

CROSS-DISCIPLINARY INTEGRATION

Work across design, animation, UI, networking, and tools so technical decisions remain practical for the full production pipeline.

UNREAL ENGINE SYSTEMS & TOOLS

ENHANCED INPUT

Input mapping and player interaction

AI PERCEPTION

Gameplay sensing and AI integration

UMG / SLATE

Runtime UI and editor-facing interfaces

MOTION WARPING

Context-aware alignment and interactions

CONTEXTUAL ANIMATION

Animation-driven gameplay interactions

MOTION MATCHING

Research and gameplay integration

MUTABLE

Runtime character customization workflows

DATA ASSETS / TABLES

Data-driven production architecture

DEVELOPMENT PRIORITIES

MAINTAINABILITY

Clear ownership, reusable APIs, long-term system health.

PERFORMANCE

Profiling-aware C++ and avoidance of unnecessary runtime cost.

MULTIPLAYER CORRECTNESS

Replication-aware design and explicit server/client responsibility.

DESIGNER ITERATION

Blueprint-facing workflows that keep implementation fast and safe.

ENGINEERING PRINCIPLES

C++ for architecture/performance-critical systems • Blueprint APIs for designer iteration • Event- and timer-driven design • Data-driven, reusable, testable systems • Production-first profiling and maintainability

CONTACT

Email • LinkedIn • Fab